باب 5

عددی نظام

(Number Systems)

5.1 ڈیٹااور انفرمیشن (Data and Information)

فیکٹس(Facts)اورفگرز(Figures) کے مجموعہ کو ڈیٹا کہتے ہیں جبکہ پروسیس کیے گئے ڈیٹا کو انفرمیشن کہتے ہیں۔

فرض کیجیے ایک جماعت کے 15 طلبا ایک امتحان میں بیٹھے ہیں۔استاد صاحب جماعت کی پاس فیصد شرح اور ہر طالب علم کا گریڈ نکالنے کو کہتے ہیں، جبکہ کل نمبرز 550 ہیں۔مطلوبہ انفرمیشن کیسے حاصل کی جائے گی؟

پہلا مرحلہ ڈیٹا جمع کرنا ہے۔ہم جماعت کے ہر طالب علم کے نمبروں کو نوٹ کرتے ہیں جو کہ یہ ہیں:

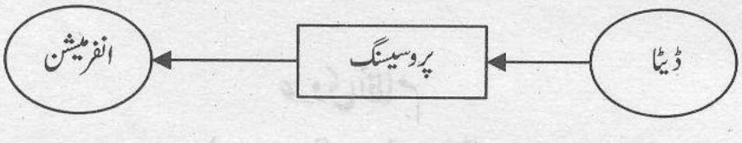
354, 285, 421, 360, 2898, 159, 163, 148, 270, 467, 305, 221, 341, 255, 311

فرض کیجیے کہ طلبا کے رول نمبرز بھی دیے گئے ہیں۔اس مرحلہ پر نمبروں کی فہرست ڈیٹا کو ظاہر کرتی ہے جو کہ طلبا نے مطلوبہ انفرمیشن حاصل کرنے کے لیے جمع کیا ہے۔یہ ڈیٹا اپنی خام شکل میں مطلوبہ انفرمیشن مہیا نہیں کرتا۔طلبا مطلوبہ انفرمیشن کو مدنظر رکھتے ہوئے اسے پروسیس کریں گے۔پروسیسنگ مختلف مراحل پر مشتمل ہوسکتی ہے۔جیسا کہ سٹورنگ، فارمیٹنگ اور خاص کیلکولیشن۔

خاص کیلکولیشن استعمال کرتے ہوئے طلبا درج ذیل جدول حاصل کرتے ہیں۔

رول نمبر	نمبرز	شرح فی صد	گریڈ	کلاس کی مجموعی شرح فی صد
1	354	64.36	B	
2	285	51.82	C	
3	421	76.55	A	
4	360	65.45	B	
5	298	54.18	C	
6	159	28.91	F	
7	163	29.64	F	
8	148	26.91	F	80%
9	270	49.09	D	
10	467	84.91	A+	
11	305	55.45	C	
12	221	40.18	D	
13	341	62.00	B	
14	255	46.36	D	
15	311	56.55	C	

درج بالا جدول ڈیٹا سے حاصل کردہ انفرمیشن کو ظاہر کرتا ہے۔یہ واضح کرتا ہے کہ اگر ڈیٹا کو ایک خاص طرح سے پروسیس کیا جائے تو ہم انفرمیشن حاصل کرتے ہیں۔پروسیسنگ کا عمل بذریعہ کمپیوٹر یا مینوئلی(Manually) سرانجام دیا جاتا ہے۔کمپیوٹر ڈیٹا کو بذریعہ ثنائی اعداد(0 یا 1) میں پروسیس کرتا ہے۔



ڈیٹا اور انفرمیشن دونوں کو بہت سی اشکال میں ظاہر کیا جاتا ہے۔ مثال کے طور پر، متن(Text)، آوازیں(Sounds)، تصاویر (Pictures)، گرافس (Graphs) وغیرہ۔ کمپیوٹر ایک پروسیسنگ مشین ہے جو ڈیٹا کو ان پُٹ کرنے کے بعد پروسیسنگ کر کے انفرمیشن آؤٹ پُٹ کرتی ہے۔ یہ ضروری ہے کہ ڈیٹا کی مختلف اقسام اور انہیں کمپیوٹر پر ظاہر کرنے کا طریقہ معلوم ہو۔

تمام کمپیوٹر پروگرامز درج ذیل میں سے کسی ایک یا ایک سے زیادہ اقسام کے ڈیٹا کو استعمال کرتے ہیں۔

(i) نومیرک ڈیٹا (ii) ایلفابیٹک ڈیٹا (iii) ایلفانومیرک ڈیٹا

5.1.1 نومیرک ڈیٹا (Numeric Data)

نومیرک ڈیٹا ان مختلف مقداروں کو ظاہر کرنے کے لیے استعمال کیا جاتا ہے جن کا حساب سے تعلق ہوتا ہے۔ مثال کے طور پر مختلف طلبا کے نمبرز، سٹور پر موجود سامان کا سیل ریکارڈ وغیرہ۔ اس ڈیٹا کو اکثر صحیح یا حقیقی اعداد کے طور پر ظاہر کیا جاتا ہے۔ مثال کے طور پر 40,323,-76.07 وغیرہ

نومیرک ڈیٹا کی دو اقسام ہیں:

(i) صحیح عدد (ii) حقیقی عدد

5.1.2 ایلفابیٹک ڈیٹا (Alphabetic Data)

یہ ڈیٹا خاص قسم کے ایلفابیٹک کریکٹرز پر مشتمل ہوتا ہے۔ مثال کے طور پر انگریزی کے بڑے حروفِ تہجی A,B,C,---,Z اور چھوٹے حروفِ تہجی a,b,c,---,z پر مشتمل ہوتا ہے۔ ہم انگریزی کے ان حروفِ تہجی کو طلبا کے نام ظاہر کرنے کے لیے استعمال کر سکتے ہیں۔ اس ڈیٹا کو کریکٹرز کی ترتیب سے ظاہر کیا جاتا ہے اور ان پر کوئی حسابی عمل نہیں کیا جاتا۔

5.1.3 ایلفانومیرک ڈیٹا (Alphanumeric Data)

یہ ڈیٹا ایلفابیٹس، اعداد اور دیگر خاص کریکٹرز جیسا کہ %, #, $ وغیرہ پر مشتمل ہوتا ہے۔ اس ڈیٹا کی مثالوں میں ٹیلی فون نمبرز اور پتے شامل ہیں۔ مثال کے طور پر (092)-041-2646916 اور مکان P.653، طارق آباد، فیصل آباد وغیرہ۔

5.2 عددی نظام (Number Systems)

عددی نظام مختلف مقداروں کو ظاہر کرنے کے لیے قیمتوں کے سیٹ کو بیان کرتا ہے۔ مثال کے طور پر ہم اپنی جماعت میں طلبا کی تعداد یا تماشائیوں کی تعداد کو جو کہ ایک خاص ٹی وی پروگرام دیکھ رہے ہوں کو ظاہر کر سکتے ہیں۔ یہ بات بہت دلچسپ اور قابلِ غور ہے کہ ڈیجیٹل کمپیوٹر میں تمام انفرمیشن اور ڈیٹا (آڈیو، گرافکس، ویڈیو، ٹیکسٹ اور نومیرک) بطور ثنائی اعداد ظاہر کیا جاتا ہے۔ عام طور پر اعشاری نظام استعمال کیا جاتا ہے جبکہ کمپیوٹر ثنائی اعداد کے نظام کو استعمال کرتے ہیں۔ عام طور پر کمپیوٹر میں اوکٹل اور ہیکسا ڈیسیمل نظام بھی استعمال ہوتے ہیں۔ یہی وجہ ہے کہ ہمیں اعداد کے ایک نظام سے دوسرے نظام میں تبدیلی کی ضرورت پیش آتی ہے۔

5.2.1 اعشاری عددی نظام (Decimal Number System)

> **کیا آپ کو یاد ہے؟**
> $10^0 = 1$
> آپ $N^0 = 1$ کے متعلق کیا جانتے ہیں، جبکہ N صفر نہیں ہے؟

ہم دس ہندسوں 0,1,2,3,4,5,6,7,8,9 سے شناسا ہیں اور ہم جانتے ہیں کہ کسی بھی قیمت کو ان دس ہندسوں کو استعمال کرتے ہوئے ظاہر کر سکتے ہیں۔ مثال کے طور پر چار سو ترپن کو درج ذیل طریقہ سے لکھا جاسکتا ہے۔

$$453 = 4\times10^2 + 5\times10^1 + 3\times10^0$$

اس نظام میں ہم کسری اعداد کو بھی لکھ سکتے ہیں۔ مثال کے طور پر 139.78 کو اعشاری نظام میں یوں لکھ سکتے ہیں:

$$139.78 = 1\times10^2 + 3\times10^1 + 9\times10^0 + 7\times10^{-1} + 8\times10^{-2}$$

> کیا آپ ایک نان پوزیشن عددی نظام کا نام بتاسکتے ہیں؟

یہ پوزیشنل عددی نظام ہے جس میں عدد کے اندر ہندسے کی پوزیشن بہت اہم ہوتی ہے۔ 39 اور 93 دو مختلف قیمتوں کو ظاہر کرتے ہیں۔

اعداد کے اس نظام میں ہر عدد ہندسوں پر مشتمل ہوتا ہے جو کہ مختلف پوزیشنوں پر مشتمل ہوتے ہیں۔ نقطہ اعشاریہ کے بائیں طرف سے پہلا ہندسہ صفر جبکہ نقطہ اعشاریہ کے بائیں طرف دوسری پوزیشن پر 1 ، اسی طرح یہ سلسلہ جاری رہتا ہے۔ اسی طرح اعشاریہ کے دائیں طرف پہلا ہندسہ پوزیشن 1- پر ہے، دوسرے کی پوزیشن 2- اور اسی طرح یہ سلسلہ جاری رہتا ہے۔ پوزیشن کی اپنی ایک اہمیت ہوتی ہے۔ مثال کے طور پر پہلی پوزیشن کا مطلب 10^0 ہے۔ دوسری پوزیشن کا مطلب 10^1 ہے اور اسی طرح یہ سلسلہ جاری رہتا ہے۔

> نوٹ کیجیے کہ پوزیشنوں کا وزن 10 کے انڈیکس میں ہے کیونکہ اعشاری نظام میں 10 اعداد ہوتے ہیں۔

اسے درج ذیل جدول سے ظاہر کیا گیا ہے۔

پوزیشن	4	3	2	1	0		-1	-2
فیس ویلیو	5	7	2	3	1		2	1
اہمیت/ وزن	10^4	10^3	10^2	10^1	10^0		10^{-1}	10^{-2}

$$57231.21 = 5\times10^4 + 7\times10^3 + 2\times10^2 + 3\times10^1 + 1\times10^0 + 2\times10^{-1} + 1\times10^{-2}$$

درج بالا مثال سے واضح ہے کہ عدد کی قیمت ہندسوں کی ضرب سے ان کی پوزیشن کے وزن کے لحاظ سے اور نتیجہ کو جمع کرنے سے بیان کی جاتی ہے۔ اس طریقہ کو پھیلاؤ کا طریقہ کہتے ہیں۔ عدد کے دائیں طرف کا انتہائی ہندسہ کم اہمیت کا ہندسہ جبکہ بائیں طرف کا انتہائی ہندسہ بہت اہمیت کا حامل ہندسہ کہلاتا ہے، کیونکہ اس کا وزن سب سے زیادہ ہوتا ہے۔ مثال کے طور پر عدد 724 میں 7 انتہائی بایاں ہندسہ ہے اور یہ سب سے زیادہ اہمیت کا حامل ہے جبکہ 4 انتہائی دایاں ہندسہ ہے اور یہ سب سے کم اہمیت کا حامل ہندسہ ہے۔

5.2.2 ثنائی عددی نظام (Binary Number System)

> ہم بائنری/ ثنائی اعداد کے پھیلاؤ میں 2 کے قوت نما کیوں استعمال کرتے ہیں؟

اس نظام میں کسی مقدار کو ظاہر کرنے کے لیے دو ہندسے صفر اور ایک (0 اور 1) استعمال ہوتے ہیں۔ ان کو ثنائی ہندسے یا بٹ (Bit) کہتے ہیں۔ اعشاری نظام کی طرح اس کو بھی پوزیشنل عددی نظام کہتے ہیں اور ہر پوزیشن کی ایک اہمیت ہوتی ہے، جو کہ 2 کی پاور ہوتا ہے۔ مثال کے طور پر

$$01001_{(2)} = 0\times2^4 + 1\times2^3 + 0\times2^2 + 0\times2^1 + 1\times2^0 = 9_{(10)}$$

پوزیشن صفر کی قدر 2^4 اور پوزیشن 1 کی قدر 2^3 ہے۔ اسی طرح ہم کسری ثنائی اعداد کو ظاہر کر سکتے ہیں۔ جیسا کہ

$$101.101_{(2)} = 1\times2^2 + 0\times2^1 + 1\times2^0 + 1\times2^{-1} + 0\times2^{-2} + 1\times2^{-3} = 5.625_{(10)}$$

جدول برائے عدد $101.101_{(2)}$

پوزیشن	2	1	0		-1	-2	-3
فیس ویلیو	1	0	1		1	0	1
وزن	2^2	2^1	2^0		2^{-1}	2^{-2}	2^{-3}

5.2.3 ہیکساڈیسیمل اعداد کا نظام (Hexadecimal Number System)

A = دس
B = گیارہ
C = بارہ
D = تیرہ
E = چودہ
F = پندرہ

ہم نے نوٹ کیا ہے کہ ثنائی اعداد کے ساتھ کام کرنا آسان نہیں ہے کیونکہ اس طرح کے اعداد کے لیے بھی کم از کم 9 بٹس درکار رہوتے ہیں یعنی $277_{(10)} = 0100010101_{(2)}$ اور اعشاری کو ثنائی میں تبدیل کرنے کے لیے کافی کیلکولیشن کرنی پڑتی ہے۔ ان مشکلات کی بناء پر کمپیوٹر سائنسدان ایک دوسرے عددی نظام کو کثرت سے استعمال کرتے ہیں۔ اس عددی نظام میں 16 مختلف ہندسے (0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F) استعمال ہوتے ہیں۔

$758_{(16)}$ ایک ہیکساڈیسیمل عدد ہے جو کہ $758_{(10)}$ سات سو اٹھاون سے مختلف ہے۔ ہم $758_{(16)}$ کو سات پانچ آٹھ بیس سولہ پڑھتے ہیں۔ یاد رہے کہ اسے سات سو اٹھاون نہیں پڑھ سکتے۔ کسری ہیکساڈیسیمل عدد $758.D1_{(16)}$ کو درج ذیل طریقہ سے ظاہر کیا جاسکتا ہے۔

$$758.D1_{(16)} = 7\times16^2 + 5\times16^1 + 8\times16^0 + D\times16^{-1} + 1\times16^{-2} = 1880.8164_{(10)}$$

چار درجہ اعشاریہ تک

جدول برائے $758.D1_{(16)}$

پوزیشن	2	1	0		-1	-2
فیس ویلیو	7	5	8		D	1
اہمیت/وزن	16^2	16^1	16^0		16^{-1}	16^{-2}

5.2.4 اوکٹل اعداد کا نظام (Octal Number System)

اس نظام کو بھی کمپیوٹر میں استعمال کیا جاسکتا ہے۔ اسے بیس (base) 8 کا یا اوکٹل عددی نظام کہتے ہیں۔ اس نظام میں صرف 8 ہندسے استعمال ہوتے ہیں جو کہ 0, 1, 2, 3, 4, 5, 6, 7 ہیں۔ اس نظام میں $751_{(8)}$ ایک مستند عدد ہے۔ یہ عدد سات سو اکاون سے مختلف ہے۔ 821 اس نظام کا عدد نہیں ہے کیونکہ 8 اس عددی نظام میں مستند ہندسہ نہیں ہے۔ اوکٹل عدد $630.4_{(8)}$ کو یوں ظاہر کیا جاتا ہے۔

$$630.4_{(8)} = 6\times8^2 + 3\times8^1 + 0\times8^0 + 4\times8^{-1} = 408.5_{(10)}$$

جدول برائے $630.4_{(8)}$

پوزیشن	2	1	0		-1
فیس ویلیو	6	3	0		4
اہمیت/وزن	8^2	8^1	8^0		8^{-1}

5.3 اعداد کے نظاموں کی تبدیلی (Conversion of Number Systems)

ہم اعشاری نظام جبکہ کمپیوٹر عام طور پر ثنائی نظام استعمال کرتے ہیں۔ کمپیوٹر کے ڈیٹا پروسیسنگ کے نظام میں اوکٹل اور ہیکسا ڈیسیمل عددی نظام کثرت سے استعمال ہوتے ہیں۔ دلچسپ مسئلہ ڈیٹا کی ایک عددی نظام سے دوسرے عددی نظام میں تبدیلی ہے۔

5.3.1 اعشاری عدد کی ثنائی عدد میں تبدیلی (Conversion of Decimal into Binary)

اعشاری عدد کو ثنائی عدد میں تبدیل کرنے کے لیے ہم بار بار تقسیم کے طریقہ کار کو استعمال کر سکتے ہیں جیسا کہ درج ذیل مثال میں دکھایا گیا ہے۔

مثال۔ 27 کو ثنائی عدد میں تبدیل کیجیے۔

حل:

	عدد	باقی
2	27	
2	13	1
2	6	1
2	3	0
2	1	1
	0	1

جب کسی تقسیم کا جواب صفر ہو تو ہمیں تقسیم کا عمل روک دینا چاہیے اور اُلٹ ترتیب سے باقی حاصل کیا جائے جیسا کہ تیر سے ظاہر ہے۔

$$27_{(10)} = 011011_{(2)}$$

5.3.2 کسری اعشاری عدد کی ثنائی عدد میں تبدیلی (Conversion of Fractional Decimal into Binary)

مثال۔ 0.56 کو ثنائی عدد میں تبدیل کیجیے۔

حل:

			نتیجہ	کسری حصہ	صحیح عدد کی حصہ
2	×	0.56	1.12	12	1
2	×	0.12	0.24	24	0
2	×	0.24	0.48	48	0

2	×	0.48	0.96	96	0
2	×	0.69	1.92	92	1
2	×	0.92	1.84	84	1
2	×	0.84	1.68	68	1
2	×	0.68	1.36	36	1

$$0.56_{(10)} = 0.10001111_{(2)}$$

5.3.3 حقیقی عداد کی ثنائی اعداد میں تبدیلی (Conversion of Real Numbers into Binary Numbers)

حقیقی اعداد کی ثنائی اعداد میں تبدیلی کے طریقہ کار کی وضاحت ایک مثال سے کی جاتی ہے۔

مثال۔ 56.25 کو ثنائی عدد میں تبدیل کیجیے۔

اس حقیقی عدد کو ثنائی میں تبدیل کرنے کے لیے ہم 56 اور 0.25 کو علیٰحدہ علیٰحدہ تبدیل کرتے ہیں۔

	عدد	باقی
2	56	
2	28	0
2	14	0
2	7	0
2	3	1
	1	1
	0	1

$$56 = 0111000_{(2)}$$

	نتیجہ	کسری حصہ	صحیح عدد کی حصہ
2 × 0.25	0.5	5	0
2 × 0.5	1.0	0	1

$$0.25 = .01_{(2)}$$

پس

$$56.25 = 0111000.01_{(2)}$$

نوٹ: مندرجہ بالا نتیجہ دونوں نتائج کو یکجا کرنے سے حاصل ہوا۔

5.3.4 ثنائی عدد کی اعشاری عدد میں تبدیلی (Convestion of Binary into Decimal)

ثنائی عدد کو اعشاری عدد میں تبدیل کرنے کے لیے پھیلاؤ کا طریقہ استعمال کیا جاتا ہے۔

مثال 1- $011011_{(2)}$ کو اعشاری عدد میں تبدیل کیجیے۔

حل: $011011_{(2)} = 0\times2^5 + 1\times2^4 + 1\times2^3 + 0\times2^2 + 1\times2^1 + 1\times2^0 = 27_{(10)}$

مثال 2- $1110.11_{(2)}$ کو اعشاری عدد میں تبدیل کیجیے۔

حل: $1110.11_{(2)} = 1\times2^3 + 1\times2^2 + 1\times2^1 + 0\times2^0 + 1\times2^{-1} + 1\times2^{-2}$

$$= 8+4+2+0+1/2+1/4$$

$$= 14.75$$

5.3.5 اعشاری عدد کی ہیکساڈیسیمل میں تبدیلی (Conversion of Decimal into Hexadecimal)

اعشاری عدد کی ہیکساڈیسیمل عدد میں تبدیلی کے طریقہ کار کی وضاحت مثالوں سے کی جاتی ہے۔

مثال 1- $185_{(10)}$ کو ہیکساڈیسیمل عدد میں تبدیل کیجیے۔

حل:

	عدد	باقی
16	185	
16	11	9
	0	B

$185_{(10)} = 0B9_{(16)}$

مثال 2- $0.3_{(10)}$ کو ہیکساڈیسیمل میں تبدیل کیجیے۔

حل:

			نتیجہ	کسری حصہ	صحیح عدد کی حصہ
16	×	0.3	4.8	8	4
16	×	0.8	12.8	8	12=C
16	×	0.8	12.8	8	12=C

پس $0.3_{(10)} = 0.4C_{(16)}$

چونکہ C تکراری قیمت ہے لہٰذا ہم اسے صرف ایک مرتبہ لیں گے۔

مثال 3- $185.3_{(10)}$ کو ہیکساڈیسیمل میں تبدیل کیجیے۔

حل: جیسا کہ اوپر والی مثالوں میں دیا گیا ہے۔

$185_{(10)} = 0B9_{(16)}$ اور $0.3_{(10)} = 0.4C_{(16)}$

لہٰذا $185.3 = 0B9.4C_{(16)}$

5.3.6 ہیکساڈیسیمل عدد کی اعشاری عدد میں تبدیلی (Conversion of Hexadecimal into Decimal)

ہیکساڈیسیمل عدد کی اعشاری عدد میں تبدیلی کی وضاحت مثالوں سے کی جاتی ہے۔

مثال 1- $0B9_{(16)}$ کو اعشاری عدد میں تبدیل کیجیے۔

حل:

$$\begin{aligned} 0B9_{(16)} &= 0\times16^2 + B\times16^1 + 9\times16^0 \\ &= 0\times16^2 + 11\times16^1 + 9\times16^0 \\ &= 185_{(10)} \end{aligned}$$

مثال 2- $0B9.4C_{(16)}$ کو اعشاری عدد میں تبدیل کیجیے۔

حل:

$$\begin{aligned} 0B9.4C_{(16)} &= 0\times16^2 + B\times16^1 + 9\times16^0 + 4\times16^{-1} + C\times16^{-2} \\ &= 0\times16^2 + 11\times16^1 + 9\times16^0 + 4\times16^{-1} + 12\times16^{-2} \\ &= 0+176+9+4/16+12/256 \\ &= 0+176+9+1/4+3/64 = 185.296875_{(10)} \end{aligned}$$

5.3.7 ہیکساڈیسیمل عدد کی ثنائی عدد میں تبدیلی (Conversion of Hexadecimal into Binary)

جیسا کہ پہلے بیان کیا گیا ہے کہ ثنائی اور اعشاری اعداد کی تبدیلی ایک مشکل طریقہ ہے جبکہ ہیکساڈیسیمل عدد کی ثنائی عدد میں تبدیلی ایک آسان طریقہ ہے جس کی وضاحت مثالوں سے کی جاتی ہے۔

مثال 1- $10.A8_{(16)}$ کو ثنائی میں تبدیل کیجیے۔

حل: عمل 1: ہر ہندسے کو ثنائی میں علیحدہ طور پر تبدیل کیجیے اور 4 بٹس میں لکھیے۔

$1 = 0001_{(2)}$

$0 = 0000_{(2)}$

$A = 1010_{(2)}$

$8 = 1000_{(2)}$

عمل 2: ہر ہندسہ کو علیحدہ علیحدہ ثنائی میں تبدیل کیجیے اور 4 بٹس میں یوں $10.A8 = 0001\ 0000\ 1010\ 1000_{(2)}$ لکھیے۔

مثال 2- A1.03 کو ثنائی میں تبدیل کیجیے۔

حل: عمل 1: ہر ہندسہ کو علیحدہ طور پر ثنائی میں تبدیل کیجیے اور 4 بٹس میں لکھیے۔

$A = 1010_{(2)}$

$1 = 0001_{(2)}$

$0 = 0000_{(2)}$

$3 = 0011_{(2)}$

عمل 2: ہر ہندسہ کو علیحدہ علیحدہ ثنائی میں تبدیل کیجیے اور 4 بٹس میں یوں $A1.03 = 1010\ 0001\ .0000\ 0011_{(2)}$ لکھیے۔

5.3.8 ثنائی عدد کی ہیکساڈیسیمل عدد میں تبدیلی (Conversion of Binary into Hexadecimal)

ثنائی عدد کی ہیکساڈیسیمل عدد میں تبدیلی مثالوں کی مدد سے واضح کی جاتی ہے۔

مثال-1 $10010011_{(2)}$ کو ہیکساڈیسیمل میں تبدیل کیجیے۔

حل:

عمل1: دائیں طرف سے شروع کرتے ہوئے دیے گئے عدد کو 4 بٹس کے گروپ میں لکھیے۔ $10010011_{(2)}$ کے دو گروپس 0011 اور 1001 ہیں۔

عمل2: ہر گروپ کو ہیکساڈیسیمل میں تبدیل کیجیے۔

$$1001 = 9_{(16)} \quad \text{اور} \quad 0011 = 3_{(16)}$$

عمل3: ہر گروپ کو اس کے متقابل ہیکساڈیسیمل میں یوں تبدیل کیجیے۔

$$10010011_{(2)} = 93_{(16)}$$

مثال-2 $101100.1_{(2)}$ کو ہیکساڈیسیمل میں تبدیل کیجیے۔

عمل1: نقطہ اعشاریہ سے شروع کرتے ہوئے دیے گئے عدد کو 4 بٹس کے گروپ میں لکھیے۔ $101100.1_{(2)}$ کے تین گروپس درج ذیل ہیں۔ 0010 اور 1100 نقطہ اعشاریہ کے بائیں طرف جبکہ 1000 نقطہ اعشاریہ کے دائیں طرف۔

عمل2: ہر گروپ کو ہیکساڈیسیمل میں تبدیل کیجیے۔

$$1000 = 8_{(16)} \quad \text{اور} \quad 0010 = 2_{(16)},\ 1100 = 12 = C_{(16)}$$

عمل3: ہر گروپ کو اس کے متقابل ہیکساڈیسیمل میں سے تبدیل کیجیے۔

$$101100.1_{(2)} = 2C.8_{(16)}$$

نوٹ کیجیے کہ اعشاریہ کے بائیں طرف آخری گروپ میں بٹس کی تعداد 4 سے کم ہے۔ اس صورت میں عدد کے انتہائی بائیں جانب زائد صفر بٹس کا اضافہ کیا جاتا ہے۔ اسی طرح اگر اعشاریہ کے دائیں طرف عدد کے آخری گروپ میں بٹس کی تعداد 4 سے کم ہو تو عدد کے انتہائی دائیں طرف صفر بٹس کا اضافہ کیا جاتا ہے۔ مندرجہ ذیل جدول کسی بھی ہیکساڈیسیمل عدد کو ثنائی عدد میں تبدیل کرنے کے لیے استعمال کیا جاسکتا ہے۔

ثنائی متقابل	ہیکساڈیسیمل عدد	ثنائی متقابل	ہیکساڈیسیمل عدد
1000	8	0000	0
1001	9	0001	1
1010	A	0010	2
1011	B	0011	3
1100	C	0100	4
1101	D	0101	5
1110	E	0110	6
1111	F	0111	7

جدول: ہیکساڈیسیمل کی ثنائی اعداد میں تبدیلی

5.3.9 اعشاری عدد کی اوکٹل عدد میں تبدیلی (Conversion of Decimal into Octal)

اعشاری عدد کی اوکٹل عدد میں تبدیلی مثالوں کی مدد سے واضح کی جاتی ہے۔

مثال-1 185 کو اوکٹل میں تبدیل کیجیے۔

حل:

	عدد	باقی
8	185	
8	23	1
8	2	7
8	0	2

$$185_{(10)} = 0271_{(8)}$$

مثال-2 $0.3_{(10)}$ کو اوکٹل عدد میں تبدیل کیجیے اور جواب 5 درجے اعشاریہ تک لکھیے۔

حل:

8×0.3	=	2.4	0.4	2
8×0.4	=	3.2	0.2	3
8×0.2	=	1.6	0.6	1
8×0.6	=	4.8	0.8	4
8×0.8	=	6.4	0.4	6

$$0.3_{(10)} = 0.23146_{(8)}$$

مثال-3 $186.3_{(10)}$ کو اوکٹل عدد میں تبدیل کیجیے اور جواب 5 درجے اعشاریہ تک لکھیے۔

حل: جیسا کہ درج بالا دونوں مثالوں سے واضح ہے کہ

$185_{(10)} = 0271_{(8)}$ اور $0.3_{(10)} = 0.23146_{(16)}$

لہٰذا $185.3_{(10)} = 0271.23146_{(8)}$

5.3.10 اوکٹل عدد کی اعشاری عدد میں تبدیلی (Conversion of Octal into Decimal)

اوکٹل اعداد کی اعشاری اعداد میں تبدیلی مثالوں کی مدد سے واضح کی جاتی ہے۔

مثال-1 $0271_{(8)}$ کو اعشاری عدد میں تبدیل کریں:

حل: $0271_{(8)} = 0\times8^3 + 2\times8^2 + 7\times8^1 + 1\times8^0 = 185_{(10)}$

مثال-2 $0271.231_{(8)}$ کو اعشاری عدد میں تبدیل کیجیے۔

حل:

$$\begin{aligned}0271.231_{(8)} &= 0\times8^3 + 2\times8^2 + 7\times8^1 + 1\times8^0 + 2\times8^{-1} + 3\times8^{-2} + 1\times8^{-3}\\ &= 0+128+56+1+2/8+3/64+1/512\\ &= 185.2988_{(10)}\end{aligned}$$

5.3.11 اوکٹل عدد کی ثنائی عدد میں تبدیلی (Conversion of Octal into Binary)

اوکٹل اعداد کی ثنائی اعداد میں تبدیلی مثالوں کی مدد سے واضح کی جاتی ہے۔

مثال-1 $107_{(8)}$ کو ثنائی عدد میں تبدیل کیجیے۔

حل: عمل1: دیے گئے عدد کے ہر ہندسے کو علیحدہ طور پر ثنائی میں تبدیل کیجیے اور تین بٹس میں لکھیے۔

$$1 = 001_{(2)}$$
$$0 = 000_{(2)}$$
$$7 = 111_{(2)}$$

عمل2: ہیکساڈیسیمل عدد کے ہندسوں کو 3 بٹس میں یوں تبدیل کیجیے۔

$$107_{(8)} = 001000111_{(2)}$$

مثال-2 $107.52_{(8)}$ کو ثنائی عدد میں تبدیل کیجیے۔

حل: عمل1: ہر ہندسے کو ثنائی میں تبدیل کیجیے اور تین بٹس میں لکھیے۔

$1 = 001_{(2)}$ $0 = 000_{(2)}$ $7 = 111_{(2)}$ $5 = 101_{(2)}$ $2 = 010_{(2)}$

عمل2: اوکٹل عدد کے ہندسوں کو تین بٹس میں یوں تبدیل کیجیے۔

$$107.52_{(8)} = 001000111.101010_{(2)}$$

5.3.12 ثنائی عدد کی اوکٹل عدد میں تبدیلی (Conversion of Binary into Octal)

ثنائی اعداد کی اوکٹل اعداد میں تبدیلی مثالوں کی مدد سے واضح کی جاتی ہے۔

مثال-1 $10010011_{(2)}$ کو اوکٹل میں تبدیل کیجیے۔

حل: عمل1: دائیں طرف سے شروع کرتے ہوئے دیے گئے عدد کو مندرجہ ذیل تین بٹس کے گروپس میں لکھیے۔

011 ، 010 اور 010

عمل2: ہر گروپ کو متقابل اوکٹل میں تبدیل کیجیے۔

$011 = 3_{(8)}$; $010 = 2_{(8)}$; $010 = 2_{(8)}$

عمل3: ہر گروپ کے لیے اس کا متقابل اوکٹل لکھیے۔

$$010010011_{(2)} = 223_{(8)}$$

مثال-2 $11010.11_{(2)}$ کو اوکٹل عدد میں تبدیل کیجیے۔

حل: عمل1: دائیں طرف سے شروع کرتے ہوئے دیے گئے عدد کو تین بٹس کے گروپس میں لکھیے۔

نقطہ اعشاریہ کے بائیں طرف 011,010 اور نقطہ اعشاریہ کے دائیں طرف 110 ہے۔

عمل2: ہر گروپ کو اوکٹل میں تبدیل کیجیے۔

$010 = 2_{(8)}$ $011 = 3_{(8)}$ $110 = 6_{(8)}$

کیا آپ 3 بٹس کے گروپ بنانے کی وجہ کا اندازہ کر سکتے ہیں؟

عمل3: ہر گروپ کا متقابل اوکٹل لکھیے۔

$$10100.11_{(2)} = 32.6_{(8)}$$

نوٹ: اگر آخری گروپ میں تین سے کم بٹس لیں تب انتہائی دائیں یا بائیں جانب بالترتیب صفر بٹس جمع کیجیے۔

جدول: اوکٹل کی ثنائی اعداد میں تبدیلی۔

اوکٹل عدد	ثنائی متقابل	اوکٹل عدد	ثنائی متقابل
0	000	4	100
1	001	5	101
2	010	6	110
3	011	7	111

5.4 1 اور 2 کے کمپلیمینٹس کے استعمال سے اعداد کا اظہار

(Representation of Numbers using 1's and 2's Complements)

علامتی اعداد کی نمائندگی (Representing Signed Numbers)

ہم جانتے ہیں کہ مثبت اعداد کو مختلف عددی نظاموں میں کیسے ظاہر کیا جاتا ہے۔ مثال کے طور پر اساس 2، اساس 10 اور اساس 16 میں۔

اب ہم ایک اور دلچسپ سوال کو دیکھتے ہیں۔

ثنائی عددی نظام میں دونوں مثبت اور منفی اعداد کو کیسے ظاہر کیا جاتا ہے؟

علامتی اعداد کو ثنائی عددی نظام میں ظاہر کرنے کے لیے بہت سے طریقے ہیں۔ مثال کے طور پر علامتی مقدار کا طریقہ، 1 اور 2 کے کمپلیمینٹس کا طریقہ اور رسائی علامت (Access notation) کا طریقہ۔ اس حصہ میں ہم 1 اور 2 کے کمپلیمینٹس کے طریقوں کو پڑھیں گے۔ یہ دونوں طریقے ثنائی حساب پڑھنے میں بڑے مفید ثابت ہوتے ہیں۔

5.4.1 1 کے کمپلیمنٹ کا طریقہ (1's Complement Method)

سب سے پہلے دیکھتے ہیں کہ کسی ثنائی عدد کے ایک کے کمپلیمنٹ سے کیا مراد ہے؟

8 بٹس کے ثنائی عدد کے ایک کے کمپلیمنٹ عدد کو $11111111_{(2)}$ میں سے تفریق کرنے سے حاصل کیا جا سکتا ہے، جیسا کہ درج ذیل مثال سے ظاہر ہے۔

مثال 1-   ثنائی عدد $(10011001)_2$ کے لیے ایک کا کمپلیمنٹ لیجیے۔

حل:

$$\begin{array}{r} 11111111 \\ -10011001 \\ \hline 01100110 \end{array}$$

ایک کے کمپلیمنٹ شکل میں

ہم دیکھتے ہیں کہ ثنائی عدد کے لیے ایک کا کمپلیمنٹ معلوم کرنے کے لیے، ہم تمام صفر کو تمام ایک اور تمام ایک کو تمام صفر میں تبدیل کر دیتے ہیں۔

مثال 2-   01100110 کے لیے ایک کا کمپلیمنٹ براہِ راست معلوم کیجیے۔

حل: دیا گیا عدد: 01100110

ایک کا کمپلیمنٹ: 10011001

منفی عدد کے ایک کا کمپلیمنٹ (Representation of Negative Numbers using 1's Complement)

کسی منفی عدد کے لیے ایک کا کمپلیمنٹ معلوم کرنے کے لیے ہم مندرجہ طریقہ اپناتے ہیں۔

☆ کسی عدد کو ظاہر کرنے کے لیے بٹس کی تعداد معلوم کریں۔

☆ دیے گئے عدد کے ماڈولس (Modulus) کو ثنائی عدد میں تبدیل کریں۔

☆ MSB میں صفر لگائیں۔

☆ نتیجہ کا ایک کا کمپلیمنٹ معلوم کریں۔

مثال 3- بذریعہ 8 بٹس $-54_{(10)}$ کو ایک کے کمپلیمنٹ سے ظاہر کیجیے۔

حل: بٹس کی تعداد = 8

$54_{(10)} = 0110110_{(2)}$

54 آٹھ بٹس شکل میں = 00110110

54- ایک کے کمپلیمنٹ شکل میں =11001001

مندرجہ بالا مثال سے ظاہر ہے کہ منفی عدد کے 1 کا کمپلیمنٹ ظاہر کرنے کے لیے MSB میں ایک ہوگا۔

5.4.2 2 کے کمپلیمنٹ کا طریقہ (2's Complement Method)

ہم جانتے ہیں کہ زیادہ تر کمپیوٹرز اعداد کو ظاہر کرنے کے لیے 16 بٹس (bits) استعمال کرتے ہیں۔ جب اعداد کو بٹس کی ایک خاص تعداد کے اندر ظاہر کیا جائے تو 2 کے کمپلیمنٹ کا طریقہ علامتی عدد کو ظاہر کرنے کے لیے بہت مفید ہوتا ہے۔ بہت سے ڈیجیٹل کیلکولیٹرز میں اعداد کو ظاہر کرنے کے لیے اس طریقہ کو استعمال کیا جاتا ہے۔

کسی ثنائی عدد کے 2 کا کمپلیمنٹ حاصل کرنے کے لیے پہلے ہم ایک کا کمپلیمنٹ حاصل کرتے ہیں اور نتیجہ میں ایک جمع کرتے ہیں۔ اس طریقہ کو درج ذیل مثال میں ظاہر کیا گیا ہے۔

مثال 1- $01100110_{(2)}$ کے لیے دو کا کمپلیمنٹ معلوم کیجیے۔

حل: عمل 1: 10011001 (دیے گئے عدد کا ایک کا کمپلیمنٹ لینے سے)

عمل 2: 10011001

+1

10011010 (ایک جمع کرنے سے)

پس عدد $01100110_{(2)}$ کا دو کا کمپلیمنٹ 10011010 ہے۔

ہم کسی عدد کا ایک کا کمپلیمنٹ لیے بغیر براہِ راست اس کا دو کا کمپلیمنٹ بھی لے سکتے ہیں۔ اس مقصد کے لیے عدد کے آخری ایک تک کوئی تبدیلی کیے بغیر صفروں کو ایک اور ہر ایک کو صفر میں تبدیل کریں۔ یہ عمل مندرجہ ذیل مثال سے واضح کیا گیا ہے۔

مثال 2- ثنائی عدد $01100110_{(2)}$ کا دو کا کمپلیمنٹ براہِ راست معلوم کیجیے۔

حل: 01100110 (دیا گیا عدد)

10011010 (دو کا کمپلیمنٹ)

منفی اعداد کا بذریعہ 2 کا کمپلیمنٹ اظہار (Representation of Negative Numbers Using 2's Complement)

ہم مندرجہ ذیل اقدام سے منفی اعداد کے لیے 2 کا کمپلیمنٹ معلوم کر سکتے ہیں۔

☆ سب سے پہلے عدد کو ظاہر کرنے کے لیے بٹس کی تعداد معلوم کریں۔

☆ ثنائی نظام میں دیے گئے عدد کے ماڈولس کو تبدیل کریں۔

☆ MSB میں صفر لگائیں اور بقیہ بٹس میں ثنائی عدد۔

☆ نتیجے کا 2 کا کمپلیمنٹ لیں۔

مثال 3- بذریعہ 8 بٹس $-54_{(10)}$ کو 2 کے کمپلیمنٹ میں لکھیے۔

حل:

بٹس کی تعداد = 8

'-54' کا ماڈولس = 54

=0110110

54 آٹھ بٹس کی شکل میں = 00110110

-54، 2 کے کمپلیمنٹ کی شکل میں = 11001010

مندرجہ بالا مثال سے ظاہر ہے کہ منفی عدد کے 2 کا کمپلیمنٹ ظاہر کرنے کے لیے MSB میں ایک ہوگا۔

بذریعہ 8 بٹس 2 کے کمپلیمنٹ میں چھوٹے سے چھوٹا عدد $= 10000000 = -128 = -2^7$

بذریعہ n بٹس 2 کے کمپلیمنٹ میں چھوٹے سے چھوٹا عدد $= 2^{(n-1)}$

5.5 ثنائی حساب (Binary Arithmetic)

اس حصہ میں، ہم ثنائی اعداد پر بنیادی حسابی عوامل یعنی جمع، تفریق، ضرب اور تقسیم سیکھیں گے۔

5.5.1 ثنائی جمع (Binary Addition)

درج ذیل جدول 2 بٹس پر جمع کے عوامل کو ظاہر کرتا ہے۔ اس جدول کو دو ضرب بِٹ ثنائی اعداد کی جمع کے لیے بھی استعمال کیا جا سکتا ہے۔

عمل	نتیجہ
0 + 0	0
0 + 1	1
1 + 0	1
1 + 1	0 1 حاصل کے طور پر

درج بالا جدول کی مدد سے مندرجہ ذیل مثال دو ثنائی اعداد کی جمع کو ظاہر کرتی ہے۔

مثال۔ $01011101_{(2)}$ اور $00110010_{(2)}$ کو جمع کیجیے۔

حل:

$$\begin{array}{r} {\scriptstyle (1)(1)(1)} \\ 01011101 \\ +00110010 \\ \hline 10001111 \end{array}$$

حاصل

> $1+1_{(2)}$ کو 2 ہونا چاہیے۔ لیکن ثنائی اعداد کے نظام میں $1+1_{(2)}$ کا جواب صفر اور حاصل ایک ہوتا ہے۔ کیا آپ بتا سکتے ہیں کہ $1+1+1_{(2)} = ?$

مندرجہ بالا مثال سے ظاہر ہے کہ دو ثنائی اعداد کی جمع کا طریقہ بھی وہی ہے جو کہ دو اعشاری اعداد کی جمع کا ہے۔ لیکن اس طریقہ میں درج بالا جدول میں دیے گئے قوانین کو استعمال کرتے ہیں۔

5.5.2 ثنائی تفریق (Binary Subtraction)

دو ثنائی اعداد کی تفریق کا طریقہ کار بھی وہی ہے جو کہ دو اعشاری اعداد کی تفریق کا ہے۔ درج ذیل جدول دو بٹس میں تفریق کے عمل کو ظاہر کرتا ہے۔

نتیجہ	عمل
0	0 - 0
1 اگلی پوزیشن سے ایک حاصل لینے سے	0 - 1
1	1 - 0
0	1 - 1

مثال1-

$$\begin{array}{r} {\scriptstyle 10\nearrow\quad\ \ 10\nearrow\qquad} \\ 10111110 \\ -01011101 \\ \hline 01100001 \end{array}$$

(حاصل)

> نوٹ: ایک اور دو کے کمپلیمنٹ کے طریقہ کو استعمال کرتے ہوئے ہم بذریعہ جمع، تفریق کر سکتے ہیں۔

ہم دیکھتے ہیں کہ جب ہم بڑی پوزیشن سے ایک حاصل لیتے ہیں تو صفر $10_{(2)}$ بن جاتا ہے اور $10_{(2)} - 1_{(2)} = 1$

وہ کمپیوٹر جو تفریق کے اس طریقہ کو استعمال کرے اس کو بنانا مشکل ہے اور اس پر لاگت بھی بہت زیادہ آئے گی۔ اس لیے زیادہ تر کمپیوٹر ایک اور دو کے کمپلیمنٹ کو تفریق کے عمل کے لیے استعمال کرتے ہیں۔

تفریق بذریعہ ایک کا کمپلیمنٹ (Subtraction Using 1's Complement)

درج ذیل مثال 8 بٹس میں بذریعہ ایک کا کمپلیمنٹ تفریق کو ظاہر کرتی ہے۔

مثال1- 38-29 کو 8 بٹس میں ایک کا کمپلیمنٹ سے حل کیجیے۔

حل: 38-29=38+(-29)

عمل1: دونوں اعداد کی مقداروں کو ثنائی شکل میں 8 بٹس کے استعمال کی مدد سے لکھیے۔

$29 = 00011101_{(2)}$ اور $38 = 00100110_{(2)}$

عمل 2: منفی اعداد کو ایک کے کمپلیمنٹ میں لکھیے۔

-29=11100010

عمل 3: ایک کے کمپلیمنٹ کو جمع کیجیے۔

00100110
+ 11100010

00001000   1: آخری حاصل

+ 1   آخری حاصل کو جمع کیجیے۔

00001001   جواب :

عمل 4: ایک کے کمپلیمنٹ کے نتیجہ کو اعشاری نظام میں تبدیل کرنے سے

00001001-9

مثال 2۔ 45-63 کو 8 بٹس میں ایک کا کمپلیمنٹ سے حل کیجیے۔

حل: 45-63-45+(-63)

عمل 1: دونوں اعداد کی مقداروں کو 8 بٹس میں لکھیے۔

$63 = 00111111_{(2)}$ اور $45 = 00101101_{(2)}$

عمل 2: منفی عدد کو ایک کا کمپلیمنٹ میں لکھیے۔

-63 = 11000000

عمل 3: ایک کے کمپلیمنٹ کو جمع کیجیے۔

00101101
+ 11000000

11101101   0: آخری حاصل

+ 0   آخری حاصل جمع کیجیے۔

11101101   جواب :

عمل 4: ایک کا کمپلیمنٹ کے نتیجہ 11101101 کو اعشاریہ میں لکھنے سے

11101101 = -00010010 = -18

نوٹ: اگر جمع میں آخری حاصل صفر ہو تو حل کا چوتھا عمل کرنے کی ضرورت نہیں۔

مثال 3۔ (-54-30) کو 8 بٹس میں ایک کا کمپلیمنٹ میں لکھیے۔

حل: -54-30=(-54)+(-30)

عمل 1: دونوں اعداد کی مقدار 8 بٹس میں لکھیے۔

$30 = 00011110_{(2)}$ اور $54 = 00110110_{(2)}$

عمل 2: دونوں اعداد کو ایک کا کمپلیمنٹ میں لکھیے۔

$-30 = 11100001$ اور $-54 = 11001001$

عمل 3: ایک کے کمپلیمنٹس کو جمع کیجیے۔

```
   11001001
+  11100001
  10101010    (آخری حاصل 1)
+         1   (آخری حاصل جمع کرنے سے)
   10101011
```

: جواب

عمل 4: ایک کا کمپلیمنٹ کے نتیجہ کو جمع کرنے سے

چونکہ MSB، 1 ہے اس لیے یہ منفی عدد ہے۔

پس 10101011 = -01010100 = -84

> نوٹ: ثنائی اعداد کی تفریق کے لیے ایک کا کمپلیمنٹ جمع کے عمل کو دو مرتبہ استعمال کرتا ہے۔ پہلے اعداد کو جمع کرتے ہوئے اور پھر آخری حاصل کو جمع کرتے ہوئے۔

تفریق بذریعہ دو کا کمپلیمنٹ (Substraction Using 2's Complement)

دو ثنائی اعداد کو بذریعہ دو کا کمپلیمنٹ تفریق کرنے کا طریقہ مندرجہ ذیل مثالوں سے واضح کیا جاتا ہے۔

مثال 1۔ 38-29 کو 8 بٹس میں دو کا کمپلیمنٹ کے طریقہ سے حل کیجیے۔

حل: 38 - 29 = 38 + (-29)

عمل 1: دونوں اعداد کی مقدار کو 8 بٹس میں لکھیے۔

$29 = 00011101_{(2)}$ اور $38 = 00100110_{(2)}$

عمل 2: منفی عدد کو دو کا کمپلیمنٹ میں لکھیے۔

-29 = 11100011

عمل 3: دو کے کمپلیمنٹ کو جمع کیجیے اور آخری حاصل چھوڑ دیجیے۔

```
   00100110
+  11100011
   00001001   (آخری حاصل 1)
```

عمل 4: دو کا کمپلیمنٹ کا نتیجہ 00001001 اعشاریہ میں تبدیل کیجیے۔

00001001 = 9

درج ذیل مثال میں ہم ایک چھوٹے عدد میں سے بڑے عدد کو تفریق کرتے ہیں۔

مثال 2- 45-63 کو 8 بٹس دو کے کمپلیمنٹ سے حل کیجیے۔

45 - 63 = 45 + (-63)

عمل 1: دونوں اعداد کی مقداریں 8 بٹس میں لکھیے۔

$63 = 00111111_{(2)}$ اور $45 = 00101101_{(2)}$

عمل 2: منفی عدد کو دو کا کمپلیمنٹ میں لکھیے۔

-63 = 11000001

عمل3: دو کے کمپلیمنٹ کو جمع کیجیے اور آخری حاصل کو چھوڑ دیجیے۔

$$\begin{array}{r} 00101101 \\ +\ 11000001 \\ \hline 11101110 \end{array}$$ (آخری حاصل صفر)

لہٰذا دو کے کمپلیمنٹ میں جواب 11101110 ہے۔

عمل4: دو کے کمپلیمنٹ کے نتیجہ کو اعشاریہ میں تبدیل کیجیے۔

11101110=-0010010=-18

نوٹ: 85-97- کو 8 بٹس دو کا کمپلیمنٹ استعمال کرتے ہوئے جواب 182- ہونا چاہیے جبکہ یہ 74 ہے۔ کیا آپ اس کی وضاحت کر سکتے ہیں؟

مثال3- 54-30- کو 8 بٹس دو کا کمپلیمنٹ کے طریقہ سے حل کیجیے۔

حل: -54-30=(-54) +(-30)

عمل1: 8 بٹس میں دونوں اعداد کی مقداریں لکھیے۔

$30 = 00011110_{(2)}$ اور $54 = 00110110_{(2)}$

عمل2: دونوں اعداد کو دو کے کمپلیمنٹ میں لکھیے۔

-54 = 11001010 اور -30 = 11100010

عمل3: دو کا کمپلیمنٹ جمع کیجیے اور آخری حاصل چھوڑ دیجیے۔

$$\begin{array}{r} 11001010 \\ +\ 11100010 \\ \hline 10101100 \end{array}$$ (آخری حاصل 1)

لہٰذا دو کے کمپلیمنٹ میں جواب 10101100 ہے۔

عمل4: دو کے کمپلیمنٹ کے نتیجہ کو اعشاریہ میں تبدیل کرنے سے

10101100 = -01010100 = -84

ہم نوٹ کرتے ہیں کہ دو اعداد کی تفریق ایک اور دو کے کمپلیمنٹ کے ذریعہ صرف جمع کے عمل کو استعمال کرتے ہوئے کر سکتے ہیں۔ اس لیے اگر کوئی دو ثنائی اعداد کو جمع کرنے سے کوئی ڈیجیٹل سرکٹ بناتا ہے تو یہی ڈیجیٹل سرکٹ دو اعداد کی تفریق کے لیے بھی کیا جا سکتا ہے۔ اسی طرح اگر ڈیجیٹل کمپیوٹر کوئی دو ثنائی اعداد جمع کر سکتا ہے تو وہ دو ثنائی اعداد کو تفریق بھی کر سکتا ہے۔

5.5.3 ثنائی ضرب (Binary Multiplication)

اس حصہ میں ہم سب سے پہلے دو غیر علامتی ثنائی اعداد کی ضرب بذریعہ عام ضرب سیکھیں گے اور اس کے بعد ایک اور دلچسپ ضرب سیکھیں گے۔

مندرجہ ذیل جدول، 2 بٹس میں بنیادی ضربی قوانین کو ظاہر کرتا ہے۔ چونکہ دو بٹس اور اعشاریہ کے درمیان غلط فہمی ممکن ہے لہٰذا اس کی جگہ × استعمال کیا جائے۔

حاصل ضرب	ضرب
0	0×0
0	1×0
0	0×1
1	1×1

مندرجہ ذیل مثال دو چار۔بٹس اعداد کے ضربی عمل کو واضح کرتی ہے۔

مثال1- $0110_{(2)} \times 1011_{(2)}$ کو حل کیجیے۔

حل:

```
    0110
  × 1011
  ------
    0110
   0110×
  0000××
 0110×××
 -------
 1000010
```

بلاشبہ یہ اعداد کی ضرب کا عمومی طریقہ ہے۔

5.5.4 ثنائی اعداد کی تقسیم (Division of Binary Numbers)

```
      01011
    --------
111) 01001101
    _00111
    ------
      1010
     _0111
     -----
       111
      _111
     -----
       000
```

ہم آسانی سے تصدیق کر سکتے ہیں کہ

$$01011_{(2)} = 11_{(10)} \text{ اور } 111_{(2)} = 7_{(10)}, 01001101_{(2)} = 77_{(10)}$$

مثال2- $01111001_{(2)} \div 1011_{(2)}$ کو حل کیجیے۔

حل:

```
       01011
     ---------
1011) 01111001
     _01011
     ------
       10000
       _1011
       -----
        1011
       _1011
       -----
        0000
```

ہم آسانی سے تصدیق کر سکتے ہیں کہ

$$01011_{(2)} = 11_{(10)} \text{ اور } 1011_{(2)} = 11_{(10)}, 01111001_{(2)} = 121_{(10)}$$

5.6 فلسڈ پوائنٹ اور فلوٹنگ پوائنٹ اعداد کا اظہار

(Fixed Point and Floating Point Number Representation)

5.6.1 فلسڈ پوائنٹ کا اظہار (Fixed Point Representation)

یہ جاننے کے لیے کہ کمپیوٹر کس طرح فلسڈ پوائنٹ کو حقیقی اعداد کے اظہار کے لیے استعمال کرتے ہیں، ہم اعداد کے اعشاری نظام کو سیکھیں گے۔

فرض کریں آپ کو مندرجہ ذیل اصولوں کے مطابق تمام حقیقی اعداد کو لکھنے کے لیے کہا جاتا ہے۔

''عدد میں نقطہ اعشاریہ سے پہلے 4 ہندسے ہوں گے اور نقطہ اعشاریہ کے بعد تین ہندسے ہوں گے''

مندرجہ ذیل جدول کا دوسرا کالم یہ ظاہر کرتا ہے کہ اس اصول کو استعمال کرتے ہوئے مختلف اعداد کو لکھا جائے گا۔

عدد	اعداد کو بذریعہ قانون لکھا گیا۔	اعداد بغیر نقطہ اعشاریہ
73.4	0073.400	0073400
120.3456	0120.345 (6 کو نہیں لکھا جا سکتا)	0120345
110	0110.000	0110000
11101.0	1101.0000 (0 کو نہیں لکھا جا سکتا)	11010000

ان اعداد کو فلسڈ (Fixed) پوائنٹ اعداد کہا جاتا ہے کیونکہ نقطہ اعشاریہ کی پوزیشن عدد کے اندر فلسڈ ہوتی ہے۔ اگر اعداد کو اس فارمیٹ (Format) میں لکھا جائے تو ہمیں نقطہ اعشاریہ لکھنے کی ضرورت نہیں ہوتی کیونکہ یہ ہمیشہ بائیں طرف سے دائیں ہندسے کے بائیں طرف ہوتا ہے۔ یہ جدول کے تیسرے کالم میں دکھایا گیا ہے۔

اس جدول سے یہ بھی واضح ہے کہ اس طرح سے تین ہندسی کسری حصہ سے زائد اور چار ہندسی صحیح حصہ سے زائد اعداد کو ظاہر نہیں کیا جا سکتا۔

حقیقی اعداد کو اس طریقے سے ظاہر کرنے کے لیے کمپیوٹر بنائے جاتے ہیں۔ کسی حقیقی عدد کو کمپیوٹر سے ظاہر کرنے کے لیے درج ذیل اصولوں کو مدنظر رکھنا ہوتا ہے۔

اعداد کو 8, 16, 32 یا زیادہ بٹس میں ظاہر کیا جا سکتا ہے، جس میں نقطہ اعشاریہ نہیں لکھا جاتا۔

- ☆ نقطہ اعشاریہ ہمیشہ دسویں بٹ کے بعد ہوتا ہے۔
- ☆ MSB کو عدد کی علامت ظاہر کرنے کے لیے استعمال کیا جاتا ہے (0 سے مراد مثبت اور 1 سے مراد منفی)۔
- ☆ اگلے بقیہ 9 بٹس کو عدد کے صحیح عددی حصہ کو ذخیرہ کرنے کے لیے استعمال کیا جائے گا۔
- ☆ بقیہ 6 بٹس کو عدد کے کسری حصہ کو استعمال کرنے کے لیے استعمال کیا جائے گا۔

اس فارمیٹ کو نیچے دکھایا گیا ہے۔

<table>
<tr><td>علامتی بٹ</td><td colspan="9">صحیح عددی حصہ</td><td colspan="6">کسری حصہ</td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
</table>

حقیقی اعداد کے اس طرح سے اظہار کو فلسڈ پوائنٹ اظہار کہتے ہیں۔ مندرجہ ذیل جدول ظاہر کرتا ہے کہ کیسے چند ثنائی اعداد کو فلسڈ پوائنٹ نمائندگی سے ظاہر کیا جاتا ہے۔

اعشاری عدد	ثنائی عدد	فکسڈ پوائنٹ شکل میں عدد
3.625	011.1010	0000000011101000
247.90625	11110111.11101	0011110111111010
-7.66796875	-0111.10101011	1000000111101010 باقی بٹس موزوں نہیں
-81.765625	-1010001.110001	1001010001110001

اس نمائندگی کو استعمال کرنے کا فائدہ یہ ہے کہ اس کو استعمال کرنا بہت آسان ہے، جبکہ نقصان یہ ہے کہ بہت چھوٹے اور بہت بڑے اعداد کو اس سے ظاہر نہیں کیا جا سکتا۔

مثال 1- 10 بٹس کو استعمال کرتے ہوئے صحیح عددی حصہ کے لیے 23.6 کو 16 بٹس فکسڈ پوائنٹ میں ظاہر کریں۔

$23 = 010111_{(2)}$ اور $0.6 = .1001100$

$23.6 = 010111.1001001 = 0000010111.100110$

فکسڈ پوائنٹ فارم $23.6 = 0000010111.100110$

مثال 2- 10 بٹس کو استعمال کرتے ہوئے صحیح عددی حصہ کے لیے 36.25- کو 16 بٹس فکسڈ پوائنٹ شکل میں لکھیے۔

$36 = 0100100_{(2)}$ اور $0.25 = .01$

$36.25 = 0100100.01 = 0000100100.01$

فکسڈ پوائنٹ فارم $36.25 = 1000100100010000$

مندرجہ ذیل مثالیں فکسڈ پوائنٹ اعداد کو اعشاری اعداد میں تبدیل کرنے کی وضاحت کرتی ہیں۔

مثال 3- فکسڈ پوائنٹ عدد 0100010111.100100 کو اعشاری عدد میں تبدیل کیجیے جبکہ صحیح عددی حصہ کے لیے 10 بٹس کو استعمال کریں۔

حل: $.100100$ = کسری حصہ اور 0100010111 = صحیح عددی حصہ

$0100010111_{(2)} = 279$

اور $.100100_{(2)} = 0.5 + .0625 = 0.5625$

پس $0100010111.100100 = 279.5625_{(10)}$

مثال 4- 16 بٹ فکسڈ پوائنٹ عدد 1000110111.110000 کو اعشاری عدد میں تبدیل کیجیے جبکہ صحیح حصہ کے لیے 10 بٹس استعمال کریں۔

حل: 110000 = کسری حصہ اور 1000110111 = صحیح عددی حصہ

$1000110111_{(2)} = -55$

اور $110000_{(2)} = 0.5 + .25 = 0.75$

لہٰذا $0100110111.110000 = -55.75_{(10)}$

5.6.2 فلوٹنگ پوائنٹ میں اظہار (Floating Point Representation)

کسی حقیقی عدد کا فلوٹنگ پوائنٹ میں اظہار ایک اور مفید طریقہ ہے۔ اس فارمیٹ میں بہت چھوٹے اور بہت بڑے اعداد کو اچھے طریقے سے ظاہر کیا جاسکتا ہے۔

$$174.592=0.174592\times10^{3}$$

اس اظہار کو سائنسی اظہار کا طریقہ کہتے ہیں، جس میں 10 اساس (Base) اور دس کی طاقت قوت نما ہے اور عدد کو مینٹیسا کہتے ہیں۔ اس طرح اوپر دیے گئے عدد میں اساس 10، مینٹیسا 0.174592 اور قوت نما 3 ہے۔ ہم ثنائی اعداد کو بھی اسی طریقہ سے لکھ سکتے ہیں۔ مثال کے طور پر

$$1000.1101=0.10001101\times2^{4}$$

یہاں اساس 2، مینٹیسا 10001101 اور قوت نما 4 ہے۔ درج ذیل پر غور کیجیے۔

مینٹیسا لکھیے	قوت نما لکھیے	علامت لکھیے

درج ذیل جدول کی مدد سے مختلف ثنائی اعداد کو درج بالا فارمیٹ کی شکل میں رکھا گیا ہے۔ نوٹ کریں کہ ثنائی پوائنٹ کو اس طرح ایڈجسٹ کیا جاتا ہے کہ تمام اعداد کے مینٹیسا میں لیڈنگ ہمیشہ 1 ہے۔

مینٹیسا	قوت نما	علامت	عدد
1.10001101	4	+	1.10001101×2^{4}
1.1101101	5	−	-1.1101101×2^{5}
1.1010011	3	+	$1101.0011=1.1010011\times2^{3}$
1.1011	−2	+	$0.01101=1.1011\times2^{-2}$

حقیقی اعداد کو لکھنے کے لیے اس فارمیٹ کو فلوٹنگ پوائنٹ کا اظہار کہتے ہیں۔ اکثر ڈیجیٹل کمپیوٹرز حقیقی اعداد کو ظاہر کرنے کے لیے اس فارمیٹ کو استعمال کرتے ہیں۔ اس کتاب میں، ہم درج ذیل فلوٹنگ پوائنٹ استعمال کریں گے۔

S	6 بٹ قوت نما						9 بٹ مینٹیسا								
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

لہٰذا کمپیوٹرز فلوٹنگ پوائنٹ اعداد کے اظہار کے لیے 16 بٹس استعمال کرتے ہیں۔

MSB جس کو S سے ظاہر کیا گیا ہے، عدد کی علامت کو ظاہر کرتا ہے۔ اگلے 6 بٹس قوت نما کو ظاہر کرنے کے لیے استعمال ہوتے ہیں جبکہ 9 بٹس عدد کے مینٹیسا کو ظاہر کرنے کے لیے استعمال ہوتے ہیں۔ اس طرح کے اظہار کے لیے اہم نقاط درج ذیل ہیں۔

☆ ثنائی فلوٹنگ پوائنٹ عدد کی علامت کو سنگل بٹ سے ظاہر کیا جاتا ہے۔ ایک بٹ منفی عدد کو اور صفر بٹ مثبت عدد کو ظاہر کرتا ہے۔

☆ قوت نما ایک علامتی صحیح عدد ہے اور اس کو 6 بٹس 2 کا کمپلیمنٹ عدد سے ظاہر کیا جاتا ہے۔

☆ مینٹیسا کا پہلا بٹ ہمیشہ 1 ہوتا ہے۔ لہٰذا اکثر نئے کمپیوٹرز میں یہ نہیں لکھا ہوتا۔

درج ذیل مثالیں اعداد کو فلوٹنگ پوائنٹ فارمیٹ میں ظاہر کرنے کے طریقہ کی وضاحت کرتی ہیں۔

مثال 1- 17.5 کو 16 بٹ فلوٹنگ پوائنٹ عدد میں ظاہر کیجیے۔

عمل 1: عدد کو ثنائی عدد میں تبدیل کیجیے۔

$$17.5=010001.10_{(2)}=1.000110\text{x}2^4$$

عمل 2: عدد کو فلوٹنگ پوائنٹ عدد میں ظاہر کیجیے۔

علامت = + = 0

قوت نما = 4

اور 6 بٹ 2 کا کمپلیمنٹ شکل میں 000100

مینٹیسا = 1.00110=1.001100000

لہٰذا فلوٹنگ پوائنٹ فارمیٹ میں عدد

S	6 بٹس قوت نما						9 بٹس مینٹیسا								
0	0	0	0	1	0	0	0	0	1	1	0	0	0	0	0

نوٹ: مینٹیسا میں پہلا ایک ثنائی عدد نہیں لکھے گئے ہیں۔

مثال 2- 117.125- کو 16 بٹس فلوٹنگ پوائنٹ عدد میں ظاہر کیجیے۔

حل: عمل 1: عدد کو ثنائی عدد میں تبدیل کیجیے۔

$$-117.125=-01110101.0010_{(2)}=-1.1101010010\times 2^6$$

عمل 2: عدد کو فلوٹنگ پوائنٹ میں تبدیل کیجیے۔

علامت = - = 1

قوت نما = 6

اور 6 بٹس 2 کا کمپلیمنٹ کی شکل میں: 000110

مینٹیسا = 1.1101010010 = 1.110101001

لہٰذا فلوٹنگ پوائنٹ فارمیٹ میں عدد درج ذیل ہے۔

S	6 بٹ قوت نما						9 بٹ مینٹیسا								
1	0	0	0	1	1	0	1	1	0	1	0	1	0	0	1

مثال3- $-0.0001101001001_{(2)}$ کو 16 بٹس فلوٹنگ پوائنٹ عدد میں ظاہر کیجیے۔

حل: $-0.0001101001001 = -1.101001001 \times 2^{-4}$

عمل2: عدد کو فلوٹنگ پوائنٹ میں تبدیل کیجیے۔

علامت = - = 1

قوت نما = 4

اور 6 بٹس 2 کا کمپلیمنٹ کی شکل میں: 111100

مینٹیسا = 1.101001001 = 1.1010010010

مزید دو مثالیں پوائنٹ عدد کی ثنائی عدد میں تبدیلی کو ظاہر کرتی ہیں۔

S	6 بٹ قوت نما						9 بٹ مینٹیسا								
1	1	1	1	1	0	0	1	0	1	0	0	1	0	0	1

مثال4- درج ذیل 16 بٹ فلوٹنگ عدد کو ثنائی عدد میں تبدیل کیجیے۔

S	6 بٹ قوت نما						9 بٹ مینٹیسا								
0	0	1	1	1	1	0	1	1	0	0	0	1	0	0	0

حل: S = 0 = +ve

قوت نما = 011110 = 30

مینٹیسا = 1.110001000

لہٰذا $1.110001000 \times 2^{30}$ ثنائی عدد ہے۔

نوٹ: مینٹیسا میں 1 بعد میں لکھا گیا ہے کیونکہ عدد کو فلوٹنگ پوائنٹ شکل میں لکھتے وقت اسے چھوڑ دیا تھا۔

مثال5- 16 بٹس فلوٹنگ پوائنٹ عدد کو ثنائی عدد میں تبدیل کیجیے۔

S	6 بٹ قوت نما						9 بٹ مینٹیسا								
1	1	0	0	1	1	1	0	1	1	1	0	1	1	0	1

حل: S = 1 = -ve

قوت نما = 100111 = - 011001 = - 25

مینٹیسا = 1.011101101

لہٰذا $-1.011101101 \times 2^{-25}$ مطلوبہ عدد ہے۔

5.7 کمپیوٹر کوڈ (Computer Code)

ہم جانتے ہیں کہ ایلفابیٹک ڈیٹا کریکٹرز پر . اور ایلفانومیرک ڈیٹا کریکٹرز اور ہندسی اعداد پر مشتمل ہوتا ہے۔ اس ڈیٹا کو کمپیوٹر میں ظاہر کرنے کے لیے ہم ایلفابیٹ کے ہر کریکٹر کے ساتھ ایک نومیرک کوڈ لگاتے ہیں۔ مثال کے طور پر A:65, B:66 وغیرہ۔ اس لیے ان کوڈز کو استعمال کرتے ہوئے ہم دونوں ایلفابیٹک اور ایلفانومیرک ڈیٹا کو کمپیوٹر سسٹم میں ظاہر کر سکتے ہیں۔

5.7.1 امریکن سٹینڈرڈ کوڈ برائے انفرمیشن انٹرچینج

ASCII (American Standard Code for Information Interchange)

ASCII ایک ایسی کوڈنگ سکیم ہے جسے آئی ایس او(ISO) نے طبع کیا ہے۔ یہ 7 بٹ کوڈنگ سکیم ہے۔ مختلف کریکٹرز کے لیے کوڈز کو درج ذیل جدول میں ظاہر کیا گیا ہے۔

جدول ASCII کوڈز

کوڈ	کریکٹر	کوڈ	کریکٹر	کوڈ	کریکٹر	کوڈ	کریکٹر
0		32	Space	64	@	96	`
1		33	!	65	A	97	a
2		34	"	66	B	98	b
3		35	#	67	C	99	c
4		36	$	68	D	100	d
5		37	%	69	E	101	e
6		38	&	70	F	102	f
7		39	'	71	G	103	g
8		40	(	72	H	104	h
9		41	)	73	I	105	i
10		42	*	74	J	106	j
11		43	+	75	K	107	k
12		44	,	76	L	108	l
13		45	-	77	M	109	m
14		46	.	78	N	110	n
15		47	/	79	O	111	o
16		48	0	80	P	112	p
17		49	1	81	Q	113	q
18		50	2	82	R	114	r
19		51	3	83	S	115	s
20		52	4	84	T	116	t
21		53	5	85	U	117	u
22		54	6	86	V	118	v
23		55	7	87	W	119	w
24		56	8	88	X	120	x
25		57	9	89	Y	121	y
26		58	:	90	Z	122	z
27		59	;	91	[	123	{
28		60	<	92	\	124	|
29		61	=	93	]	125	}
30		62	>	94	^	126	~
31		63	?	95	_	127	

نوٹ: 0-31 کوڈز کے لیے کوئی کریکٹر نہیں ہے۔

ASCII میں کریکٹر A اور a کے لیے مختلف کوڈ ہیں۔ اکثر کمپیوٹر 8 بٹ ASCII کوڈز بھی استعمال کرتے ہیں۔ 8-بٹ ASCII کوڈز میں بقیہ 128 کوڈز گرافکل اور دوسرے خاص کریکٹرز ظاہر کرنے کے لیے استعمال ہوتے ہیں۔ درج ذیل مثالیں مختلف پیغامات کے لیے ASCII کے استعمال کو ظاہر کرتی ہیں۔

مثال 1- Binary کو بذریعہ ASCII ظاہر کیجیے۔

حل: ASCII کوڈ کے استعمال سے ہم دیکھتے ہیں کہ

کریکٹر	اعشاری کوڈ	ثنائی کوڈ
B	66	01000010
i	105	01101001
n	110	01101110
a	97	01100001
r	114	01110010
y	121	01111001

اس طرح ہم Binary کو یوں لکھ سکتے ہیں:

01000010 01101001 01101110 01100001 01110010 01111001

مثال 2- درج ذیل ASCII پیغام کو انگلش میں تبدیل کیجیے۔

01010111 01101000 01100001 01110100 00111111

ASCII کو استعمال کرتے ہوئے ہم دیکھتے ہیں:

ثنائی کوڈ	اعشاری کوڈ	کریکٹر
01010111	87	W
01101000	104	h
01100001	97	a
01110100	116	t
00111111	63	?

01010111 01101000 01100001 01110100 00111111

پیغام What? کو ظاہر کرتا ہے۔

5.7.2 ثنائی کوڈڈ اعشاریہ (Binary Coded Decima-lBCD)

اس کوڈنگ سکیم کو نومیرک ڈیٹا ظاہر کرنے کے لیے استعمال کیا جاتا ہے۔ ہم جانتے ہیں کہ اعشاری عددی نظام میں دس ہندسے ہوتے ہیں۔ ان ہندسوں کو ظاہر کرنے کے لیے ہمیں 4 بٹ کوڈز کی ضرورت ہوتی ہے۔ BCD میں ہندسوں کے ساتھ درج ذیل کوڈز لگائے جاتے ہیں۔

BCD کوڈز کا جدول

کوڈ	ہندسہ	کوڈ	ہندسہ	کوڈ	ہندسہ	کوڈ	ہندسہ	کوڈ	ہندسہ
0100	4	0011	3	0010	2	0001	1	0000	0
1001	9	1000	8	0111	7	0110	6	0101	5

درج ذیل مثال BCD میں غیر منفی صحیح اعداد کو ظاہر کرتی ہے۔

مثال - 9807 کو BCD میں ظاہر کیجیے۔

حل: ہم جانتے ہیں کہ BCD میں

9 = 1001,

8 = 1000,

0 = 0000,

اور 7 = 0111

لہٰذا 9807 = 1001 1000 0000 0111

واضح طور پر ہمیں 4 ہندسی عدد کو ظاہر کرنے کے لیے 16 بٹس کی ضرورت ہوتی ہے۔ اسی عدد کو ثنائی میں 14 بٹس استعمال کرتے ہوئے ظاہر کر سکتے ہیں۔ BCD کوڈز زیادہ بٹس استعمال کرتے ہیں۔ لہٰذا کمپیوٹر میموری کی مزید ضرورت ہوتی ہے۔

وہ اعداد جو BCD میں کوڈز ہوتے ہیں اُن پر حسابی عوامل کرنے کے لیے یا تو انہیں پہلے ثنائی میں تبدیل کرنا پڑتا ہے اور تب حسابی عوامل کرتے ہیں یا اس مقصد کے لیے خاص سرکٹ ڈیزائن کرنے پڑتے ہیں۔

5.7.3 توسیعی بائنری کوڈڈ ڈیسیمل انٹرچینج کوڈ

(Extended Binary Coded Decimal Interchange Code-EBCDIC)

IBM نے ایک نئی کریکٹر کوڈنگ سکیم متعارف کروائی ہے جسے EBCDIC کہتے ہیں۔ یہ موجودہ BCD کوڈز کی طرح کی بہتر سکیم ہے۔ یہ 8 بٹ کوڈ ہے، لہٰذا EBCDIC میں 256 مختلف کوڈ ظاہر کیے جاسکتے ہیں۔ یہ کثرت سے استعمال ہونے والے کریکٹر کوڈز تھے لیکن پرسنل کمپیوٹر کے بڑھتے ہوئے استعمال اور کمپیوٹر نیٹ ورکس کی بناء پر ASCII کوڈنگ سکیم ایک سٹینڈرڈ کوڈنگ سکیم بن گئی ہے اور اب اکثر کمپیوٹرز ASCII استعمال کرتے ہیں۔

درج ذیل جدول چند کریکٹرز اور EBCDIC کوڈز کو ظاہر کرتا ہے۔

مختلف کریکٹرز کے لیے EBCDIC کوڈز کا جدول

ہیکس کوڈ	کریکٹر	ہیکس کوڈ	کریکٹر	ہیکس کوڈ	کریکٹر	ہیکس کوڈ	کریکٹر
C0	{	D0	}	E0	\	F0	0
C1	A	D1	J	E1	.	F1	1
C2	B	D2	K	E2	S	F2	2
C3	C	D3	L	E3	T	F3	3
C4	D	D4	M	E4	U	F4	4
C5	E	D5	N	E5	V	F5	5
C6	F	D6	O	E6	W	F6	6
C7	G	D7	P	E7	X	F7	7
C8	H	D8	Q	E8	Y	F8	8
C9	I	D9	R	E9	Z	F9	9

5.7.4 یونی کوڈ (Unicode)

ان دونوں استعمال ہونے والی کوڈنگ سکیموں میں یونی کوڈ ایک مقبول کوڈنگ سکیم ہے۔ یہ 16 بٹ کوڈنگ سکیم ہے، اس لیے اس سکیم میں زیادہ کریکٹرز کو ظاہر کیا جاسکتا ہے۔

مشق

1۔ درج ذیل کی وضاحت کیجیے۔

(a) ثنائی اعداد کا نظام (b) اوکٹل اعداد کا نظام (c) اعشاری اعداد کا نظام

(d) ہیکساڈیسیمل اعداد کا نظام (e) ASCII کوڈز (f) BCD

2۔ مثالوں کی مدد سے درج ذیل کی وضاحت کیجیے۔

(a) ڈیٹا (b) انفرمیشن

3۔ مختلف کمپیوٹر ایپلیکیشنز میں استعمال ہونے والے ڈیٹا کی بڑی اقسام کون سی ہیں؟ ان کی وضاحت کریں اور ان پر لاگو عوامل بیان کریں۔

4۔ علامتی اعداد کو ظاہر کرنے کے لیے 2 کا کمپلیمنٹ کے طریقہ کار کی وضاحت کیجیے۔ اس طریقہ کو استعمال کرتے ہوئے ہم تفریق کا عمل کیسے کر سکتے ہیں؟

5۔ علامتی اعداد کو ظاہر کرنے کے لیے 2 کا کمپلیمنٹ کے طریقہ کی وضاحت کریں۔ اس طریقہ سے تفریق کے عمل کی وضاحت کریں۔

6۔ درج ذیل اعشاری اعداد کو ثنائی، اوکٹل اور ہیکساڈیسیمل میں تبدیل کیجیے۔

(a) 78 (b) 97 (c) 129

7۔ درج ذیل ہیکساڈیسیمل اعداد کو ثنائی، اوکٹل اور اعشاری اعداد میں تبدیل کیجیے۔

(a) $7A_{(16)}$ (b) $1C2_{(16)}$ (c) $89_{(16)}$

8۔ درج ذیل اوکٹل اعداد کو ثنائی، اعشاری اور ہیکساڈیسیمل اعداد میں تبدیل کیجیے۔

(a) $125_{(8)}$ (b) $57_{(8)}$ (c) $777_{(8)}$

9۔ ذیل ثنائی اعداد کو اوکٹل، اعشاری اور ہیکساڈیسیمل میں تبدیل کیجیے۔

(a) $01110101_{(2)}$ (b) $10101001_{(2)}$ (c) $00110011_{(2)}$

10۔ دیے گئے BCD اعداد کو ڈیسیمل 10011001 میں تبدیل کیجیے۔

(a) 10101001 (b) 00000111 (c) 10000001

11۔ درج ذیل اعداد کو 8 بٹ 1 کمپلیمنٹ اور 10 بٹ 2 کا کمپلیمنٹ اعداد میں تبدیل کیجیے۔

(a) 76 (b) -98 (c) -126

12۔ درج ذیل 8 بٹ 1 کمپلیمنٹ اعداد کو اعشاری اعداد میں تبدیل کیجیے۔

(a) 00101011 (b) 10001001 (c) 11111111

13۔ درج ذیل 8 بٹ 2 کا کمپلیمنٹ اعداد کو اعشاری اعداد میں تبدیل کیجیے۔

(a) 00111101 (b) 11111111 (c) 10101010

14۔ 8 بٹ 1 کا کمپلیمنٹ کا طریقہ استعمال کرتے ہوئے درج ذیل تفریق کیجیے۔ جواب کی تصدیق اعشاری اعداد میں تبدیل کر کے کریں۔ تمام اعداد اعشاری نظام میں ہیں۔

(a) 127-126 (b) 12-106 (c) -12-25

15- 8 بٹ 2 کا کمپلیمنٹ کا طریقہ استعمال کرتے ہوئے درج ذیل تفریق کیجیے۔ جواب کی تصدیق کو اعشاری اعداد میں تبدیل کر کے کیجیے۔
تمام اعداد اعشاری نظام میں ہیں۔

(a) -57-96 (b) -120-110 (c) -60-68

16- 10 بٹ 1 کا کمپلیمنٹ اور 2 کا کمپلیمنٹ کا طریقہ استعمال کرتے ہوئے درج ذیل تفریق کیجیے۔ نتیجہ کی تصدیق کے لیے اپنے جواب کو اعشاریہ میں تبدیل کیجیے۔

(a) -57-96 (b) -120-110 (c) -60-68

17- 8 بٹس میں چھوٹے سے چھوٹا اور بڑے سے بڑا عدد کیا ہے؟

18- 8 بٹس 1 کمپلیمنٹ میں چھوٹے سے چھوٹا اور بڑے سے بڑا عدد کیا ہے؟

19- 8 بٹس 2 کا کمپلیمنٹ میں چھوٹے سے چھوٹا اور بڑے سے بڑا عدد کیا ہے؟

20- درج ذیل اعداد کو فِکسڈ پوائنٹ سے ظاہر کیجیے۔ تبدیلی کے لیے درج ذیل فارمیٹ استعمال کیجیے۔ اپنے نتیجہ کی تصدیق کے لیے نتیجہ کو واپس اعشاری اعداد میں تبدیل کیجیے۔

(a) 25.5 (b) 233.9 (c) 33.6

10 بٹس انٹیگرل حصہ کے لیے	6 بٹس کسری حصہ کے لیے

21- درج ذیل اعداد کو فِکسڈ پوائنٹ اظہار استعمال کرتے ہوئے ظاہر کیجیے۔ تبدیلی کے لیے سابقہ سوال میں دیا گیا فارمیٹ استعمال کیجیے۔ کسی مشکل کی صورت میں وضاحت بھی کیجیے۔

(a) 1025.5 (b) 1233.9 (c) 2333.6

22- درج ذیل اعداد کو فلسڈ پوائنٹ استعمال کرتے ہوئے ظاہر کیجیے۔ باب میں دیے گئے فلوٹنگ پوائنٹ فارمیٹ کو استعمال کیجیے۔

(a) 1025.5 (b) 1233.9 (c) 2333.6

23- درج ذیل پیغامات کو ASCII کوڈز کو استعمال کرتے ہوئے ظاہر کیجیے۔ اپنے کوڈ پیغام کو واپس انگلش میں تبدیل کرتے ہوئے تصدیق کیجیے۔ (سپیس کریکٹر کو تبدیل کرنا نہ بھولیے)

(i) He is a good student

(ii) 2+2=4

(iii) I like Computer Science

(iv) Binary numbers are GREAT

24- خالی جگہ پُر کیجیے۔

(i) _____ غیر مترتب اعداد و شمار ہیں جن کی پروسیسنگ سے انفرمیشن حاصل ہوتی ہے۔

(ii) پروسیس کیا گیا ڈیٹا _____ کہلاتا ہے۔

(iii) _____، _____ اور _____ علامتی اعداد کے اظہار کے تین طریقے ہیں۔

(iv) ASCII سے مراد ------- ہے۔

(v) $1\ 000\ 0100\ 0010 = (\quad)_{16}$

(vi) $1\ 000\ 100\ 010 = (\quad)_{8}$

(vii) $00100011_{(2)}$ کا 2 کا کمپلیمنٹ ------ ہے۔

(viii) کمپیوٹر ہر چیز کو ------ کی شکل میں مینوپلیٹ کرتا ہے۔

(ix) ہیکساڈیسیمل عدد کی اساس ------ ہے۔

(x) ------ میں آخری حاصل (end carry) کو ختم کر دیا جاتا ہے۔

25۔ درج ذیل کو ملائیے۔

ASCII	ڈیٹا
$22_{(10)}$	پروسیسنگ
پروسیس کیا گیا ڈیٹا	انفرمیشن
غیر مترتب اعداد و شمار جن کا کوئی مطلب نہیں اور جو پروسیسنگ کے لیے تیار ہیں۔	ASCII
پروسیسنگ سے مراد ڈیٹا کو مینوپلیٹ کرنا، کیلکولیٹ کرنا، تقسیم کرنا یا ترتیب دینا ہے۔	$16_{(16)}$
$22_{(8)}$	$12_{(16)}$

26۔ درست جواب لکھیے۔

(a) ہیکساڈیسیمل عدد $10_{(16)}$ کے برابر ہے۔

(i) $10_{(10)}$ (ii) $100_{(10)}$ (iii) $16_{(10)}$ (iv) پہلے تمام

(b) ہیکساڈیسیمل عدد $100_{(16)}$ کے برابر ہے

(i) $0001\ 0000\ 0000_{(2)}$ (ii) $256_{(10)}$ (iii) $400_{(8)}$ (iv) پہلے تمام

(c) $0101010_{(2)}$ کا 2 کا کمپلیمنٹ ہے۔

(i) 1010110 (ii) 1010101 (iii) 0000011 (iv) ان میں سے کوئی نہیں

(d) منفی بائنری عدد کا 1 کا کمپلیمنٹ حاصل کیا جاتا ہے۔

(i) عدد میں بٹس کو اُلٹنے سے (ii) عدد میں بٹس کو اُلٹنے سے اور ایک جمع کرنے سے

(iii) کیلکولیٹ نہیں کیا جا سکتا (iv) (i) اور (ii) دونوں

(e) (011) 4752105 ہے۔

(i) نومیرک ڈیٹا (ii) ایلفانومیرک ڈیٹا (iii) ایلفابیٹک ڈیٹا (iv) دونوں (i) اور (ii)

27۔ درج ذیل میں درست اور غلط کی نشاندہی کیجیے۔

(i) ایسا کمپیوٹر بنانا جو کہ اعشاری عددی نظام استعمال کرتا ہو، ناممکن ہے۔

(ii) $1234_{(16)} = 11011100_{(2)}$

(iii) دنیا میں تمام کمپیوٹرز ASCII کوڈز استعمال کرتے ہیں۔

(iv) 1 اور 2 کا کمپلیمنٹ کا طریقہ بٹس کی مخصوص تعداد کے لیے قابل عمل ہے۔

(v) ہم 256 کا اظہار بذریعہ 8 بٹس نہیں کر سکتے۔

(vi) اوکٹل عددی نظام میں کُل 8 بنیادی ہندسے ہیں۔

(vii) ASCII 7 بٹس کوڈنگ سکیم ہے۔

(viii) یونی کوڈ سافٹ ویئر میں ملٹی لینگوئل مدد مہیا کرنے کے لیے استعمال ہوتا ہے۔

(ix) BCD سے مراد بائنری کوڈڈ ہندسے ہیں۔

(x) ہیکساڈیسیمل عددی نظام میں G کی قیمت 16 کو ظاہر کرتی ہے۔

جوابات

24. (i) ڈیٹا (ii) انفرمیشن (iii) 2's کمپلیمنٹ، 1's کمپلیمنٹ، سائنڈ مقدار، (iv) امریکن سٹینڈرڈ کوڈ فار انفرمیشن انٹرچینج
(v) 1024 (vi) 1024 (vii) 11011101 (viii) بائنری عدد (ix) 16 (x) 2's کمپلیمنٹ

26. (a) iii (b) iv (c) i (d) i (e) ii

27. (i) F (ii) F (iii) F (iv) T (v) T
(vi) T (vii) T (viii) T (ix) F (x) F